

Electronic Assignment Cover sheet

Please fill out and attach as the first page of Assignment.

Student (s) Number as per your student card:

<u>Robin Varma</u>	<u>10631497</u>
<u>Rohit Varma</u>	<u>10635420</u>
<u>Supritha Sridhar Rao</u>	<u>10639851</u>

Course Title: MSc in Business Analytics

Lecturer Name: Basel Magableh

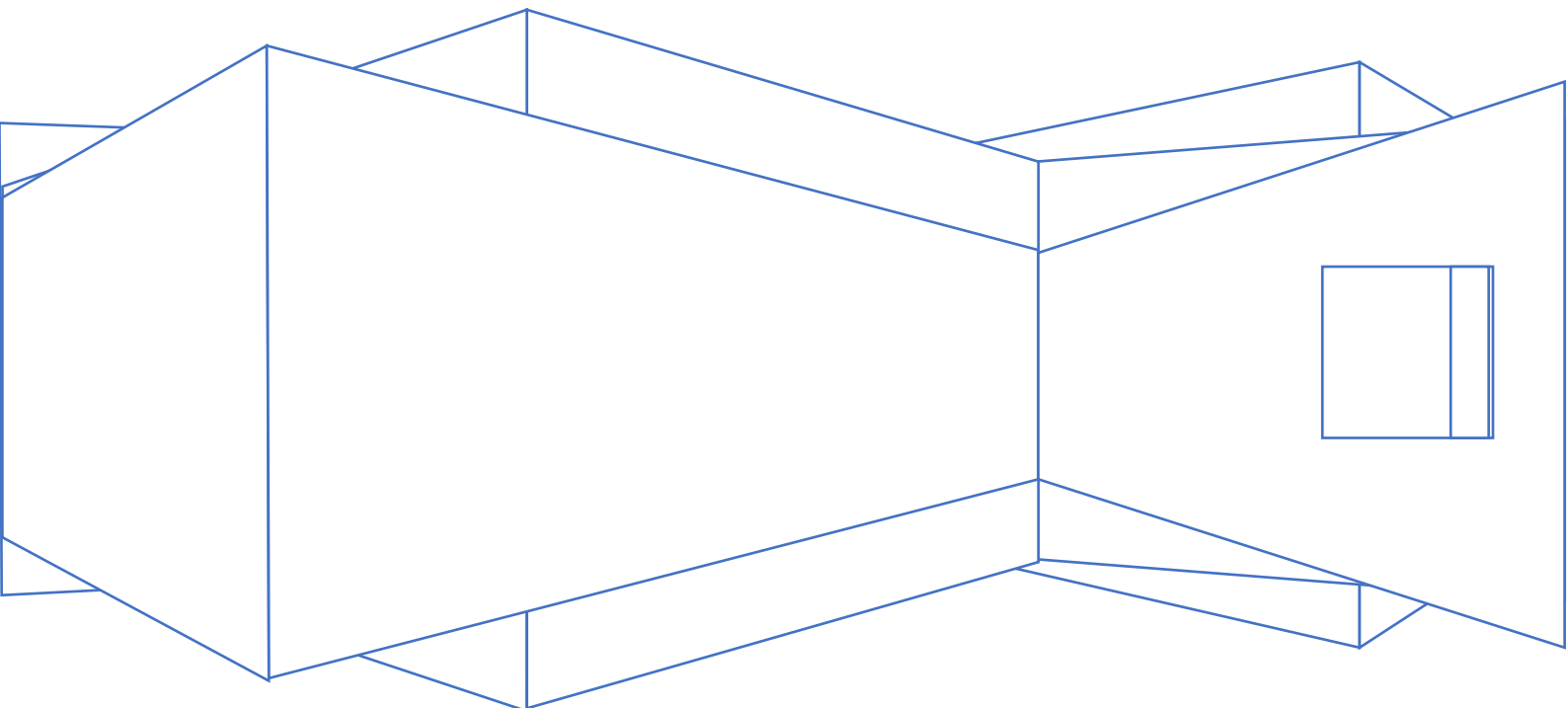
Module/Subject Title: Requirement Analysis

Assignment Title: Requirement Analysis

No of Words:

Note: Technical support is available to student between 0930- 1700 hrs only. There is no technical support after 1700 hrs. It is your responsibility to ensure that you allow time to troubleshoot any technical difficulties by uploading early on the due date.

REQUIREMENT ANALYSIS REPORT



Group Assignment Details

- + Module name : Requirement Analysis
- + Title of the Design : Clever Par-King Application
- + Group name : Clever king

Student IDs along Student ID

- + Robin Varma - 10631497
- + Rohit Varma - 10635420
- + Supritha Sridhar Rao - 10639851

Table of Contents

1.1 Project Title	5
1.2 Overview.....	5
1.3 Business Requirements	6
1.4 Agile Epics and User Stories.....	9
2.Solution	9
3.Strategy.....	10
4. Payment.....	13
1.5 User Case model.....	15
1.6 Application User Interface.....	17
1.7 Backend Operations modal	18
1.8 Conclusion	20

1.1 Project Title

Our project is named - **Clever Par-king (Park like a king)**

1.2 Overview

We live in a world that is connected, moves quickly and constantly getting smarter, where cities, retails, offices and transportation all runs on data. Despite this rapid progress traffic and parking patterns has remain large constant since the invention of the autos the result, visitors come to the expense centre and queue to the ticket window to pick up the parking ticket, after that look for a parking space, which is solely luck leading to unnecessary congestion and emissions, stress of circling around and wasting of time and with increasing population and vehicles this will become a serious problem in the coming years.

After doing some research, we discovered that there will be a parking issue in Ireland's major cities within the next ten years. The government is most concerned about this issue because there is currently no clever solution due to the rising rate of private vehicle usage in urban areas as a result of rapid economic and population growth. We wanted to capitalize on people's urge to use their smartphones at all hours of the day. In order to make cities smarter, we chose to tackle the parking problem by fusing it with the internet. We made the decision to develop this parking software, which is simple to use on smartphones and gives you real-time data using our overhead sensor installed at parking lots for stadiums, movie theatres, markets, and other parking-related locations.

Clever Par-King is a mobile application which gives users smart parking solutions with the help of real time data, providing turn by turn guidance to the most convenient parking space which will reduce a large amount of time and stress of a driver, making the uncertainty of parking a thing of the past from outdoor to indoor to on-street parking. Generating accurate data in real time about each available parking space and transforms that data into efficient

guidance for drivers. Data is gathered through overhead sensors that will cover up to 100 spaces per single sensor, drivers are then guided to available parking spaces seamlessly on mobile app. This sensor will collect real time data and via router it will be stored in then certain cloud system and the client with mobiles can access the system to see the interface of every parking available with the colour coordination like, if the place is marked with red the parking is occupied and if it is green the parking is available.

Ground rules of our team are:

- Our team is structured democratically. There is no permanent leader on our team. Every decision is made by team consensus.
- The ground rules for our team stipulate that we must meet every Monday and that everyone must be on time.
- He/she is required to give a justification if he /she can't make it to the meeting.
- The tasks allocated on the date of the previous meeting should be completed before the date of the succeeding meeting.
- Everyone on the team needs to routinely check her mail.
- How emergency meetings are handled is decided by the team.

1.3 Business Requirements

We've done extensive research in two key areas. First, we have done market research on mobile applications that are comparable to our product as well as on social media platforms where we might find inspiration or ideas. The second component of our research—technological analysis—helped us choose the platforms, tools, APIs, and libraries we would employ. Analysis, initial design, detailed design, prototype release, implementation, testing, and maintenance are all phases that our team is expected to move through. It is an Android Application and to develop it, UI development, Google Map API and Android SDK will be used

If a User wants to use this app, he/she will have opened the free *Clever Par-king app*, register (optional) and the app is ready to launch. Whenever the user

about to go outside, they can search in advance if parking is available or not, select the parking of choice based on location, timing, pricing , photos and parking spot according to the car size. The app will give real time and predicted update of the parking peak hours, then do advance reservation (optional), the app then navigates user turn by turn directly to the parking reducing stress of circling around and wasting of time. After reaching the destination the user will see an indicator at the entrance of the parking displaying the availability. The parking attendant at the entrance with the help of mobile based digitized ticketing system gives the user a printed ticket and the barriers will open turning green and opens up , the user can also use there digital ticket or a smart card or they can put our smart sticker with QR code on it which can also be used for payment (in case when reservation is not done) at the entrance, the QR code scanner and the sensors on the barrier will detect the licence plate and QR code gathering details and storing in it (for security and safety purpose) and the barriers will open itself, keeping in and out time stamp. Then the intelligent parking guiding system will give the user direction to the right spot.

All this data will be gathered by sensors and cameras will be send to a router and from there it will be send and stored in a cloud server. This real time information then is sent to the parking management on our mobile app and web dashboard to the government authorities on a citywide parking dashboard and to all the citizens using Clever Par-king app. When the user comes back and leaves the parking spot the sensors will sense it and at the exit the right amount of parking fee deducted automatically from user's mobile wallet or smart card or Smart sticker or can pay in cash or credit card to our smart parking attendant or an auto pay station, the barriers will open and they can leave without hassle.

Requirement Analysis -

1. Requirement

During this stage, all potential system needs are gathered and outlined in a requirement specification document. The results of surveys and interviews were used to determine user demands, such as the ability to check the availability of parking spaces, reserve one, and make payments.

2. *Design*

In this phase, the required specifications from the initial phase are examined, and the system design is created. This system design aids in determining the overall system architecture as well as the hardware and system requirements. The following phase is to develop the system that was created after completing surveys to determine each user's demands. UML diagrams were employed in the design of this investigation. Choosing the tools to build the system comes next, starting with creating logical processes and a display that all users can understand. The primary programming language used to create this software is react native because it is appropriate for creating mobile-based programs.

3. *Coding and Unit Test*

The system is initially built as small programs known as units, which are then incorporated into the following phase, using inputs from the system design. Unit testing was done in process for developing and evaluating each unit for functionality. The major phase, primarily the coding, we used java script language to build the mobile platform.

4. *Analysis Data and Information Need*

Data needed to make smart parking applications are parking capacity data, vehicle volume, and parking rates. While the information that needs to be generated is information about the amount of parking capacity available, how to place an order, and payment of parking. The analysis made is this Information System using a flow map, which provides an overview of the processes that occur in the application so that the application can find information and find out easily.

4. *Maintenance*

The client environment can provide certain problems. Patches are published to address certain problems. Additionally, improved versions of the product are issued. To bring about these changes in the surroundings of the consumer, maintenance is performed.

1.4 Agile Epics and User Stories

1.User Research

We performed user interviews, previous surveys, and researches on parking issues to validate the problem, gain a deeper understanding of the problems experienced by the users in their everyday lives, and to foster user empathy, I identified the main problems that users are having after conducting the research and analysing the data I gathered along the way. The following are the main issues, as reported by the users:

- "I have a hard time finding parking close to where I want to go."
- "Sometimes, I'm not sure if there's enough parking."
- "There have been times when I have lost track of where my car is parked in the crowded parking space."
- "I have no choice but to park here, even though some places are not safe to do so."
- "It can be challenging to determine whether a parking space is authorized or not."
- "Paying cash makes me uncomfortable, especially now that Covid is in power."
- "Parking can be very expensive sometimes, but I never find out the cost until I get to the parking."

2.Solution

We created the Clever Park-King mobile app with the user's needs and business objectives in mind to create a seamless and hassle-free experience for everyone who struggles to find parking spots for their automobiles.

In my mind's eye, a map that may be used to guide users to adjacent parking spots.

I came up with ideas for the app's functionality before moving on to create the user flow and wireframes. These characteristics were present:

- ✓ Choose the fastest path to the nearby parking lots to help you navigate.
- ✓ Research parking costs and availability in advance.
- ✓ Freedom to choose and reserve a parking space in advance.
- ✓ Time slots can be extended
- ✓ Pay digitally through cards, wallet transfer, Smart Stickers etc
- ✓ Find out the authorization status and operating hours of the neighbouring parking spaces.
- ✓ Easily find the parked vehicle

3.Strategy

User Flow

We created the user flow, which outlines how the user will interact with the app and the processes necessary to reserve the first parking spot, based on the features we wanted to include in the app. We made sure that there were only a few simple steps that the users would be able to follow.

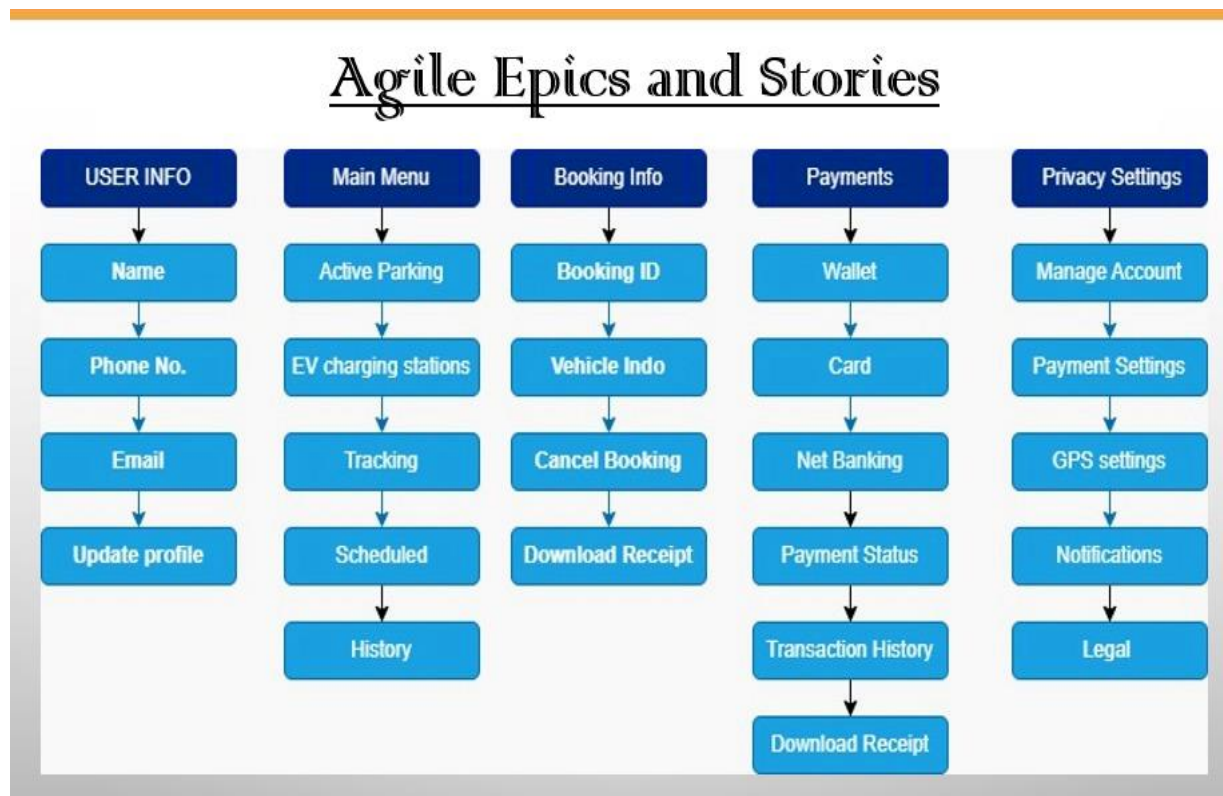
1.Onboarding

Simple but eye-catching interfaces that communicate the app's value proposition to users.

2. User Parking Flow

The displays that consumers will interface with for the following are listed below:

- I. *Look for or choose a parking spot close by.*
- II. *Determine the cost and available parking spaces in advance.*
- III. *Locate the parking space and choose the length of time to stay.*
- IV. *Pre-reserve a parking space,*
- V. *Enter your arrival time, pay electronically, and take advantage of hassle-free parking.*



1. Onboarding Process

The first time a user sees your app is during onboarding.

You may see the displays for both new and existing users by looking below. The new user must submit his basic personal information as well as the car he will be driving. The vehicle's model (name) and number are its details.

For some reason, you removed the app from your device. When you return, installing it once more will give you two choices. Given the circumstances, you will pick "existing user" because you are aware that you already have an account.

We designed the onboarding process such that it clarifies the purpose of the app for users. We kept it straightforward and modest so that the user wouldn't have to navigate through several screens.

2. The four main actions are searching, choosing a vehicle, navigating, and scanning.

Users can enter the parking location on the Home Screen. If the user accesses the parking space either by searching or by setting a location on the map, I thought it was simple.

The search results will now provide parking pins. Users can tap on any of the pins to get information about time, location, price, and other things.

Map: The map shows the user's current location and nearby parking spaces. It also allows users to visualize the location of different parking spaces and choose the one that best suits their needs.

Filter: The filter option allows users to refine their search based on specific criteria such as price, distance and availability.

Choose the SUV, Prime, Sedan, EV or Non-EV category for the vehicle you are driving. The parking hub will make use of that information.

Navigating was the following step. We set limitations, such as traveling 5 km to the parking area.

Parking lot information: When users click on a parking lot on the map, they can see information such as price, distance to destination, and availability.

At the entry, RFID scanning happens when the user arrives at their parking destination. The user can access parking only after that. The payment method will be accessed using the same scanning procedure.

3.Track a parked car

We frequently witness folks losing track of where they parked. This issue would be resolved by adding tracking. It also informs you of how long you have left your car parked.

4. Payment

We decided to display the payment screen after the user has finished scanning. With information on the parked duration and area, we have given the pricing in a more focused manner, which is significant.

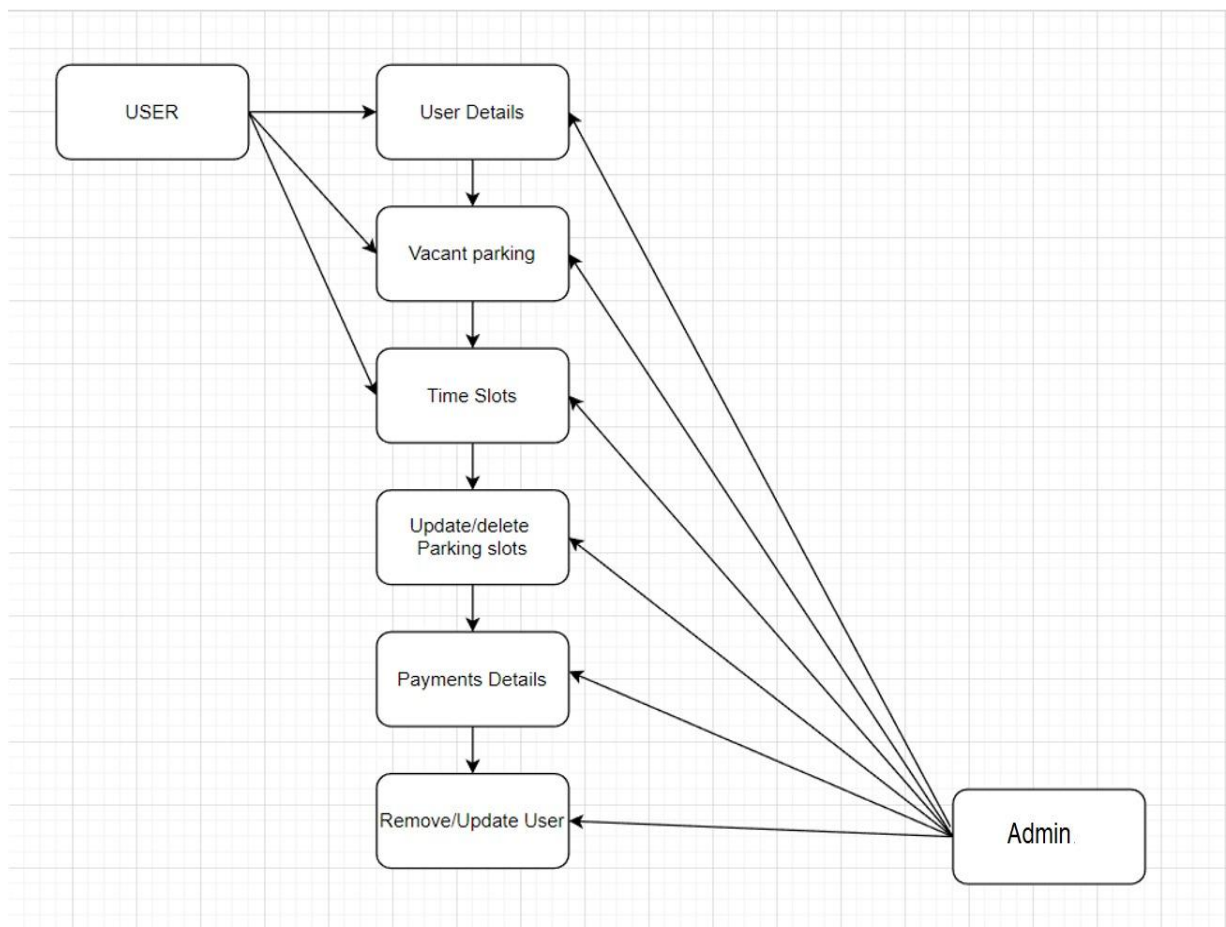
Online payments are simple and can be made with a card, wallet, or net banking. The payment success screen will then appear to let the user know the transaction was successful.

Remaining screens there will be a drop down containing-

1. User Info- Displaying users name, digital picture, phone number, Email and more
2. Main Menu - Containing information about the vehicles booking, track, etc.
3. Vehicle update - The user can update its vehicle as in can delete or add a new vehicle etc.
4. Payment- Showing information of Wallet, Card, Net banking, history/cancellation, download etc.
5. Privacy Settings- Containing account and payment settings with notification option and more

After payment is complete, the user must offer comments. To see how useful the app is.

1.5 User Case model



The image you provided is a mock-up interface design for a parking app. The interface is split into two sections: one for the user and the other for the moderator.

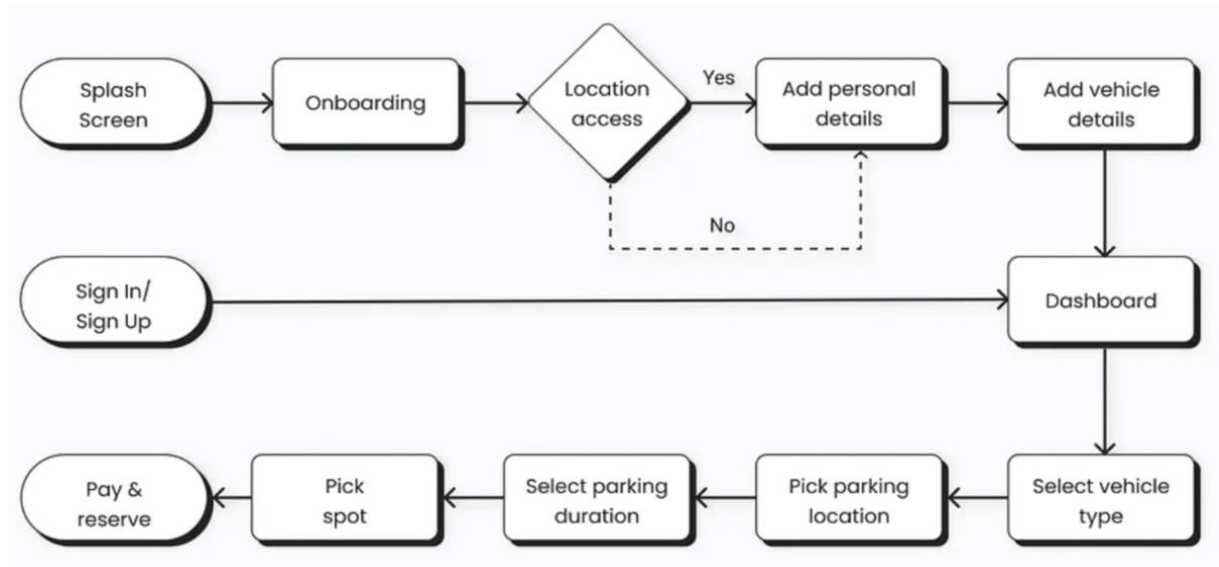
The user section is on the left side of the interface. It includes a map of the parking area, with pins indicating the available parking spots. The user can select a parking spot by clicking on the pin. The selected parking spot will be highlighted on the map and information about the spot, such as the parking rate and the hours of operation, will be displayed in a box above the map. The user can then choose to reserve the parking spot by clicking on the "Reserve Now" button.

The user section also includes a "My Reservations" tab, which shows the user's current and upcoming parking reservations. The user can cancel or modify their reservations from this tab.

The moderator section is on the right side of the interface. It allows the moderator to manage the parking spots and reservations. The moderator can view the current reservations, approve or reject reservation requests, and manage the parking rates and hours of operation. The moderator can also add or remove parking spots from the map.

Overall, this interface provides an easy-to-use platform for users to reserve parking spots and for moderators to manage the parking area. It is designed to be intuitive and user-friendly, with clear visual cues and easy-to-navigate menus.

1.6 Application User Interface



Multipurpose parking app that enables users to easily navigate and manage their parking experience is shown above.

A search bar located at the top of the screen enables users to look for parking lots or garages close to their current location or a particular destination. If you need to find a parking spot quickly and efficiently in a new city or area but don't know the layout well, this feature can be especially helpful.

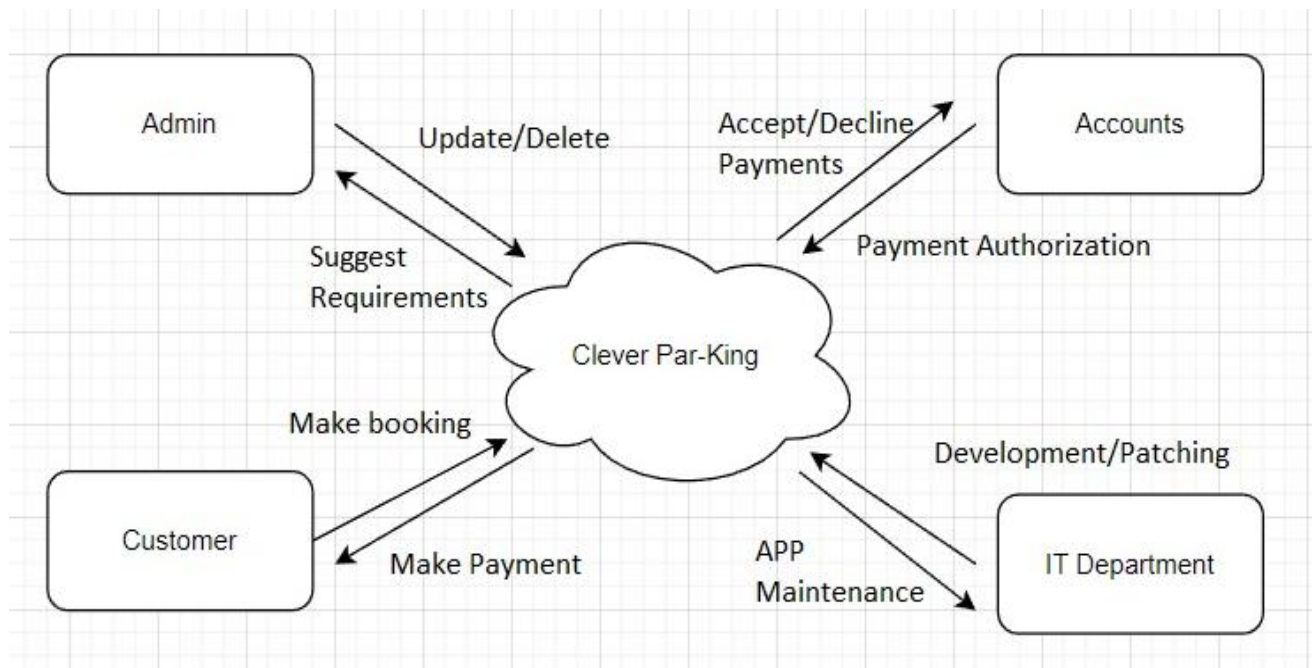
A map displaying the user's current location and accessible parking options is located beneath the search bar. The color-coded map shows how many parking spots are available at each location. Red denotes that there is no more space in the garage or lot, while green denotes that parking is free. By assisting users in finding open spaces in parking lots quickly, this feature can help users save time and frustration.

The app also enables users to reserve parking spots in advance, which can be helpful in areas with high traffic or during events where parking may be difficult to come by. When making a reservation, users can choose the desired parking space, the date, and the time in the app.

The ability to pay for parking directly through the app is another helpful feature. The app calculates the total cost after users enter their license plate number and the length of time they parked. Because you don't need to look for and pay for each individual meter, this feature can save users time and effort. .

Overall, this parking app offers a variety of features that make it simple for users to navigate and manage their parking experience, from looking for open parking spaces to reserving spots and paying for parking.

1.7 Backend Operations modal



The image you provided shows the architecture of the backend operation of a mobile app. Here is an explanation of each component:

Mobile app: This is the application that runs on the user's mobile device. It communicates with the backend server to perform various tasks such as data retrieval, processing, and storage.

Load balancer: The load balancer is responsible for distributing incoming traffic evenly across multiple backend servers. This ensures that no single server becomes overwhelmed with traffic and helps to maintain the performance and availability of the system.

Web server: The web server is a software application that receives and processes HTTP requests from the mobile app. It serves web pages or data back to the app through HTTP responses. It also interacts with the application server to retrieve and process data.

Application server: The application server is responsible for executing the business logic of the mobile app. It processes requests from the web server and interacts with the database server to retrieve and store data.

Database server: The database server is where data for the mobile app is stored. It manages the storage and retrieval of data and provides a structured way to organize and store the app's data.

Cache server: The cache server is used to temporarily store frequently accessed data to reduce the amount of time it takes to retrieve data from the database server. This can help to improve the performance of the mobile app.

Overall, the backend operation of a mobile app involves several components that work together to provide a reliable and efficient service to the end-user. The load balancer, web server, application server, database server, and cache server all play a critical role in ensuring that the mobile app is fast, reliable, and secure.

1.8 Conclusion

The many kinds of smart parking systems are discussed in this paper. The effectiveness of the smart parking system in reducing traffic, particularly in urban areas where there is traffic congestion and a lack of parking places, is evident from the different examples of its application that are being provided. It accomplishes this by guiding customers and making the best use of parking spaces. One of the most important components of the smart parking system, the study on all the sensor technologies used in detecting vehicles

When the threshold distance was calibrated and the obstruction was recognized, the smart auto parking system that was created, made, and tested produced accurate results. Automation is a positive step toward a successful future in the transportation industry.

It may be inferred that an automatic smart automobile parking system can be made, which would save pointless driving, wasteful use of fuel, and time, as well as make parking a lot easier, with the proper connection of a few basic electrical components.

We found it to be quite fulfilling to work on the design process, which is mostly based on the user needs and pain points identified through research. I gained various insights into how goods are truly built via exploring applications.

The feedbacks which we got from users made me think and do changes in designs accordingly. Altogether process was challenging and fun. The system is expected to facilitate visitors because they do not have to look for parking spaces suddenly and can find out the status and information on the available parking. The outcome of all the long hours spent trying to develop a minimal and easy to use app led to a potential solution that feels like it was always meant to be there.